Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 292 (2026) 35–44

Procedia Computer Science

www.elsevier.com/locate/procedia

30th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2026)

Multilingual Energy–Performance Evaluation of Large Language Models

I. de Zarzà[a], Mauro Liz[b,c], J. de Curtò[b,d,*], Jordi Cabot[a,e], Carlos T. Calafate[f]

[a]*Human Centered AI, Data & Software, LUXEMBOURG Institute of Science and Technology, L-4362 Esch-sur-Alzette, Luxembourg*
[b]*Escuela Técnica Superior de Ingeniería (ICAI), Universidad Pontificia Comillas, 28015 Madrid, Spain*
[c]*Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215, USA*
[d]*Department of Computer Applications in Science & Engineering, BARCELONA Supercomputing Center, 08034 Barcelona, Spain*
[e]*Université du Luxembourg, L-4365 Esch-sur-Alzette, Luxembourg*
[f]*Departamento de Informática de Sistemas y Computadores, Universitat Politècnica de València, 46022 València, Spain*

Abstract

The rapid adoption of Large Language Models (LLMs) in practical applications has increased the importance of understanding their inference-time energy consumption, particularly in multilingual and large-scale deployments. While LLMs are typically evaluated using accuracy-centric benchmarks, energy efficiency is often treated as a secondary concern and rarely analyzed jointly with performance across languages. In this work, we present a systematic energy–performance evaluation of five instruction-tuned LLMs across thirteen languages under controlled inference conditions. Using a fixed problem set and identical hardware configuration, we measure GPU-level energy consumption together with multiple performance indicators, including overall task score, step accuracy, and consistency. Our results reveal substantial and statistically significant differences in energy consumption across models (Friedman χ^2=49.42, p=4.78×10^{-10}), with up to threefold variation in watt-hours per 1,000 queries under identical workloads. Importantly, we observe that higher energy consumption does not consistently translate into improved reasoning performance (pooled Spearman r_s=0.11, p=0.39). Going beyond aggregate per-query energy figures, we additionally normalize energy by the number of tokens actually generated and find that this reordering completely changes the efficiency ranking: compact, optimization-oriented models retain a substantial per-token energy advantage, whereas short-output models lose their apparent edge once output length is controlled. Furthermore, we show that language choice has a measurable impact on both energy usage and model rankings, leading to language-dependent efficiency frontiers and Pareto-optimal configurations. Among the evaluated models, compact reasoning-oriented architectures achieve a more favorable balance between performance and energy demand, whereas larger or more energy-intensive models often fail to deliver proportional gains in accuracy. These findings highlight the need for energy-aware and multilingual evaluation practices and support the use of AI energy scores as a complementary criterion for selecting LLMs in sustainable, real-world inference scenarios.

© 2026 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0)
Peer-review under responsibility of the scientific committee of the KES International Association

Keywords: Large Language Models; Energy Efficiency; Multilingual Evaluation; Sustainable AI; GPU Energy Consumption; AI Energy Scores

* Corresponding author. Tel.: +34 934 13 70 18.

1877-0509 © 2026 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0)
Peer-review under responsibility of the scientific committee of the KES International Association

1. Introduction and Related Work

Large Language Models (LLMs) have rapidly become a core component of modern intelligent systems, enabling advanced reasoning, natural language understanding, and decision support across a wide range of applications [1, 2, 3]. Their integration into real-world deployments, such as conversational agents, decision-support platforms, and multi-agent systems [4], has shifted attention from purely algorithmic performance toward operational considerations, including computational cost, latency, and energy consumption. As LLM-based systems scale in both usage and geographic reach, energy efficiency has emerged as a first-order concern, particularly in the context of sustainable and resource-aware AI systems [5, 6].

Despite this growing relevance, LLM evaluation practices remain predominantly accuracy-centric. Benchmarks typically emphasize task success rates, aggregated performance scores, or reasoning accuracy [7, 8], while inference-time energy consumption is either omitted or treated as a secondary constraint [9]. This evaluation gap is especially pronounced in multilingual settings, where LLMs are increasingly expected to operate across diverse languages under identical infrastructure constraints. In such scenarios, the assumption that energy consumption scales uniformly with model size or correlates directly with performance is rarely validated empirically.

Recent advances in model architectures have further complicated this picture. Compact and optimization-focused designs aim to deliver competitive reasoning capabilities at reduced computational cost [10, 11], while larger or more expressive models often incur substantially higher energy demands [6, 12]. Understanding how these architectural choices translate into performance-per-energy outcomes is essential for informed model selection, particularly in production environments where operational efficiency, sustainability, and cost predictability are critical. However, systematic and statistically grounded analyses that jointly consider energy consumption, task performance, and multilingual behavior remain limited [13].

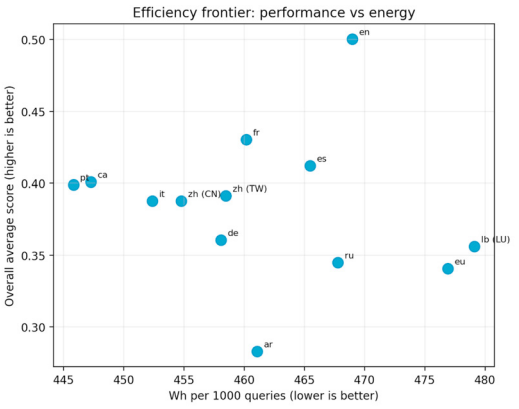
Furthermore, recent work has revealed that larger and more instructable language models do not always exhibit improved reliability [14], and that consistency across semantically equivalent inputs remains a significant challenge [15, 16]. These findings motivate a more nuanced approach to LLM evaluation that considers not only peak performance but also operational characteristics such as energy efficiency and behavioral stability.

In this work, we address this gap by presenting a controlled, multilingual energy–performance evaluation of five instruction-tuned Large Language Models across thirteen languages. Using a fixed problem set and a consistent hardware configuration, we measure GPU-level energy consumption alongside multiple performance indicators, including overall task score, step accuracy, and consistency. This paired experimental design allows us to isolate model- and language-dependent effects while ensuring comparability across all configurations.

The contributions of this paper are threefold: (1) we provide a systematic multilingual benchmark that jointly evaluates performance and energy consumption of LLMs under controlled conditions; (2) we quantify the extent to which energy efficiency and reasoning performance diverge across models and languages; and (3) we highlight the importance of energy-aware evaluation frameworks for sustainable deployment of LLM-based systems. While prior work on AI energy scores and efficiency comparisons has typically been demonstrated on simpler tasks or limited evaluation settings [9, 12], our work is among the first to examine energy–performance trade-offs on multilingual, complex reasoning tasks using controlled measurements and multi-objective analysis [17]. Together, these contributions support a shift from accuracy-only benchmarks toward multi-objective evaluation practices that better reflect the operational realities of modern AI systems.

The remainder of this paper is organized as follows. Section 2 describes the experimental setup, including the evaluated models, the multilingual dataset and its translation provenance, hardware configuration, performance metrics, and the statistical framework. Section 3 presents the main findings, covering energy consumption across models, task performance outcomes, language-dependent effects, the new per-token analysis, and Pareto frontier analysis. Section 4 interprets the results in the context of sustainable LLM deployment and discusses limitations. Section 5 summarizes the key contributions.

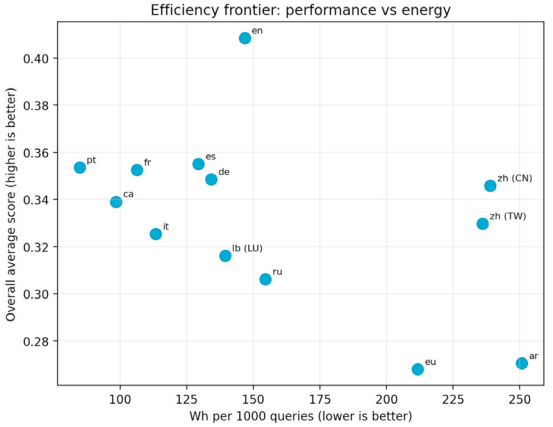
E-mail address: jdecurto@icai.comillas.edu



Energy performance by language using `falcon-mamba-7b`.

language	gpu_total_wh	wh_per_1000_queries	gpu_avg_watts	gpu_max_watts	wall_time_sec	wh_per_prompt
en	37.05	468.99	236.06	246.14	564.97	0.4690
ar	36.42	461.07	235.67	245.49	556.40	0.4611
ca	35.33	447.27	235.72	246.96	539.60	0.4473
de	36.19	458.09	236.35	246.57	551.20	0.4581
es	36.77	465.49	236.35	246.52	560.08	0.4655
eu	37.68	476.90	236.48	245.98	573.49	0.4769
fr	36.35	460.18	238.92	249.45	547.68	0.4602
it	35.74	452.39	239.49	249.18	537.18	0.4524
lb	37.85	479.12	239.16	248.07	569.66	0.4791
pt	35.22	445.83	240.51	250.21	527.14	0.4458
ru	36.96	467.79	241.84	251.66	550.03	0.4678
zh-cn	35.93	454.79	241.37	250.51	535.78	0.4548
zh-tw	36.22	458.48	241.82	250.82	539.17	0.4585

Fig. 1: Energy–performance frontier for Falcon-Mamba-7B across languages. Lower-right regions indicate higher efficiency.



Energy performance by language using `Mistral-7B-Instruct-v0.1`.

language	gpu_total_wh	wh_per_1000_queries	gpu_avg_watts	gpu_max_watts	wall_time_sec	wh_per_prompt
en	11.59	146.73	311.17	339.47	134.07	0.1467
ar	19.81	250.81	317.83	340.47	224.41	0.2508
ca	7.77	98.40	304.03	337.47	92.04	0.0984
de	10.60	134.17	311.50	340.94	122.47	0.1342
es	10.22	129.41	309.02	339.04	119.08	0.1294
eu	16.73	211.72	315.07	340.15	191.10	0.2117
fr	8.40	106.30	305.51	337.41	98.95	0.1063
it	8.95	113.34	305.89	337.97	105.33	0.1133
lb	11.01	139.41	310.75	341.95	127.58	0.1394
pt	6.70	84.75	299.67	336.09	80.43	0.0848
ru	12.20	154.48	309.39	339.07	141.98	0.1545
zh-cn	18.88	238.97	317.47	339.28	214.03	0.2390
zh-tw	18.65	236.07	318.49	339.35	210.73	0.2361

Fig. 2: Energy–performance frontier for Mistral-7B-Instruct-v0.1. The model occupies the low-energy, low-performance extreme.

2. Materials and Methods

We evaluate five instruction-tuned Large Language Models representing different architectural and optimization strategies: Falcon-Mamba-7B, Mistral-7B-Instruct-v0.1, Phi-3-mini-128k-instruct, Phi-4-mini-instruct, and Qwen2-7B-Instruct (see Table 1 for detailed specs). These models were selected to cover a range of reasoning capabilities, parameter efficiencies, and deployment profiles commonly encountered in contemporary LLM-based systems [2, 1].

Table 1: Model specifications and architectural characteristics.

Model	Params (B)	Architecture	Context (K)	FP16 Weights (GB)
Falcon-Mamba-7B	7.27	SSM (Mamba)	8†	~14
Mistral-7B-Instruct-v0.1	7.24	Transformer+GQA	8	~14
Phi-3-mini-128k-instruct	3.82	Transformer	128	~8
Phi-4-mini-instruct	3.84	Transformer+GQA	128	~8
Qwen2-7B-Instruct	7.07	Transformer+GQA	32	~14

†Training context length; Mamba-based models do not impose a fixed attention window at inference.
GQA = Grouped-Query Attention; SSM = State Space Model.
FP16 denotes half-precision (16-bit) floating-point model weights. Reported sizes include only model parameters and exclude KV cache, activations, optimizer state, and runtime overhead.

All models were executed using the same vLLM 0.5.0 HTTP endpoint [18] with PyTorch 2.3.0 [19] as backend, without task-specific fine-tuning, to isolate architectural and implementation-level differences. Decoding used low-temperature sampling (temperature = 0.2, max_tokens = 300), batch size 1 per API call, and three independent runs per problem. This evaluation focuses exclusively on inference-time behavior, which is the dominant cost factor in many production deployments [6].

The evaluation dataset consists of a fixed set of 79 reasoning-oriented problems, first introduced in [20, 17], translated into thirteen languages: Arabic (ar), Catalan (ca), German (de), English (en), Spanish (es), Euskera (eu), French (fr), Italian (it), Luxembourgish (lb), Portuguese (pt), Russian (ru), Simplified Chinese (zh-cn), and Traditional Chinese (zh-tw). Each language variant preserves the semantic structure and difficulty of the original problems, enabling paired comparisons across languages. The multilingual set was produced via a three-stage pipeline: automated `deep-translator` drafts, LLM post-editing for terminological and numerical consistency, and human verification by native or near-native speakers, as in [17].

By using an identical problem set across all models and languages, the experimental design controls for task variability and allows language-specific effects on both performance and energy consumption to be analyzed directly. This multilingual setup reflects realistic deployment scenarios in which LLMs are expected to operate consistently across diverse linguistic contexts [21], and addresses concerns about the predominance of English-centric evaluation in prior benchmarking efforts [13].

All experiments were conducted on an NVIDIA H200 GPU to eliminate hardware-induced variability. GPU power consumption was monitored throughout execution using NVIDIA's management library, and energy usage was computed by integrating power measurements over time. The same batch size, decoding strategy, and execution parameters were applied across all runs.

Energy metrics include total GPU energy consumption (Wh), watt-hours per 1,000 queries, average and peak GPU power, and wall-clock execution time. These measurements capture both instantaneous power behavior and cumulative energy cost, which are critical for evaluating sustainability in large-scale deployments [12, 9]. Our measurement approach follows recommendations for systematic reporting of energy consumption in machine learning systems [9].

Our measurement approach aligns with the AI Energy Score framework [22], which reports GPU consumption in watt-hours per 1,000 queries under controlled conditions. We adopt this as our primary energy metric and define performance efficiency as the ratio of overall task score to E_{1000}, enabling direct comparison of sustainability–performance trade-offs across models.

Model performance was assessed using three complementary metrics: overall task score $S_{\text{overall}} \in [0, 1]$ (end-to-end success rate), step accuracy $S_{\text{step}} \in [0, 1]$ (intermediate-reasoning correctness, following chain-of-thought evaluation practices [10, 23]), and consistency C. Consistency is formally defined as the mean per-problem standard deviation of scores across the three independent runs: $C = \frac{1}{N}\sum_{o=1}^{N}\sigma_o$, $\sigma_o = \text{std}(s_o^{(1)}, s_o^{(2)}, s_o^{(3)})$, where $s_o^{(k)}$ is the score of the k-th run on the o-th problem and N=79. Lower C indicates greater output stability under repeated low-temperature sampling. This empirical-stability metric complements the consistency-as-similarity formulations of [15, 11].

These metrics were chosen to reflect different dimensions of reasoning quality and to avoid over-reliance on a particular performance indicator. Recent work has shown that single-metric evaluations can obscure important behavioral differences across models [13, 8]. All metrics were computed uniformly across models and languages to ensure fairness and comparability.

Performance–energy trade-offs were analyzed using Pareto frontier analysis, identifying configurations that jointly minimize energy consumption and maximize performance. This multi-objective perspective, based on principles of dominance and non-dominated sorting [24], provides actionable insight into model selection under competing constraints.

3. Results

Substantial differences in energy consumption were observed across the evaluated models, despite identical hardware, workloads, and execution parameters.

Friedman tests with the thirteen languages as paired blocks confirm highly significant differences across models on all metrics: energy (χ^2=49.42, p=4.78×10^{-10}), overall score (χ^2=46.58, p=1.86×10^{-9}), step accuracy (χ^2=43.32, p=8.87×10^{-9}), and consistency (χ^2=29.66, p=5.74×10^{-6}).

As shown in Table 2, Mistral-7B-Instruct-v0.1 consistently exhibits the lowest energy consumption, with average values below 160 Wh per 1,000 queries across languages. In contrast, Falcon-Mamba-7B and Qwen2-7B-Instruct operate at significantly higher energy levels, exceeding 450 Wh per 1,000 queries on average. The Phi-family models occupy an intermediate regime, with Phi-4-mini-instruct achieving the lowest energy consumption among high-performing models. Pairwise Wilcoxon signed-rank tests with Holm correction confirm that the three energy regimes

Table 2: Cross-model summary of energy consumption and performance metrics (mean ± std across 13 languages). The AI Energy Score is defined as the ratio between overall task score and watt-hours per 1,000 queries, scaled by 10^3 for readability. Models are ordered by AI Energy Score (higher is better).

Model	Wh/1000 queries	Overall	Step Acc.	AI En. Score ($\times 10^3$)
Mistral-7B-Instruct-v0.1	157.3 ± 57.4	0.33 ± 0.04	0.39 ± 0.03	2.10
Phi-4-mini-instruct	251.3 ± 8.6	0.51 ± 0.07	0.66 ± 0.08	2.03
Phi-3-mini-128k-instruct	272.1 ± 6.3	0.48 ± 0.07	0.65 ± 0.07	1.76
Qwen2-7B-Instruct	464.6 ± 7.9	0.47 ± 0.05	0.64 ± 0.06	1.01
Falcon-Mamba-7B	461.3 ± 10.2	0.38 ± 0.05	0.51 ± 0.05	0.82

(low: Mistral; mid: Phi-3, Phi-4; high: Falcon-Mamba, Qwen2) are statistically separable at α=0.05, with paired Cohen's d effect sizes between regimes uniformly large ($|d| > 2.0$), e.g. Phi-4 vs. Falcon-Mamba $d = -23.59$, Mistral vs. Falcon-Mamba $d = -5.49$. The only model pair not separable in energy is Qwen2 vs. Falcon-Mamba (p_{Holm}=0.31, d=0.32).

Table 2 summarizes the mean energy consumption and performance metrics across all thirteen languages, illustrating the approximately threefold difference in energy demand between the most and least efficient models.

Energy–performance frontiers reveal distinct efficiency profiles. Falcon-Mamba-7B and Mistral-7B-Instruct-v0.1 achieve lower overall scores (Figures 1 and 2), with Mistral occupying the lowest performance tier across languages while also defining the low-energy extreme of the evaluation space. Phi-3-mini-128k-instruct offers competitive performance with notable stability (Figure 4), while Phi-4-mini-instruct achieves the highest overall performance while maintaining moderate energy usage, occupying a favorable region of the energy–performance frontier (Figure 5). Qwen2-7B-Instruct delivers competitive reasoning accuracy but does not outperform the Phi models despite substantially higher energy consumption (Figure 6). Pairwise Wilcoxon Holm comparisons confirm that Phi-4 significantly outperforms Falcon-Mamba and Mistral on overall score (d=2.44 and 3.79 respectively, p_{Holm}=0.0024), while the difference between Phi-4 and Phi-3 is smaller in magnitude ($d = -0.86$, p_{Holm}=0.018), and Qwen2 does *not* significantly outperform Phi-3 on overall score ($d = -0.25$, p_{Holm}=0.31) despite consuming 70% more energy.

Language choice has a measurable impact on both energy consumption and performance efficiency. Figure 8 presents model-by-language heatmaps showing deviations from the English baseline. While energy consumption varies moderately across languages for a given model, performance efficiency exhibits larger language-dependent fluctuations. As a result, model rankings are not stable across languages. Configurations that are near-optimal in one language may become dominated in others, particularly when evaluated jointly on energy and performance metrics. This observation underscores the limitations of English-only benchmarking when assessing real-world multilingual deployment scenarios.

Output-length-normalized energy: a per-token view

The per-query metric E_{1000} has an under-recognized confound: it integrates the energy cost of every generated token, so models that produce shorter responses appear more efficient even when their per-token cost is high. Because autoregressive generation cost scales approximately linearly with the number of decoded tokens, normalizing energy by the actual number of generated tokens disentangles *architectural* per-token efficiency from *operational* verbosity-driven cost. This decomposition is acknowledged but not measured in [17]; we resolve it here.

For each model–language configuration we tokenize the full set of generated responses (79 problems × 3 runs) recovered from the raw evaluation logs, using a cross-lingual UTF-8-bytes/4 proxy (the per-model tokenizers are not loaded post hoc, but the proxy is widely used in the literature [25] and preserves cross-model and cross-language ratios up to a multiplicative scaling). We then compute $E_{1000\text{tok}} = E_{\text{total}}/(T/1000)$, where T is the total generated-token count.

Table 3 summarizes generation verbosity and per-token energy. The picture changes substantially relative to Table 2. Mistral-7B produces dramatically shorter outputs (mean ≈ 64 tokens/response) than the other models, which explains a large part of its apparent per-query advantage; once output length is controlled, Mistral's per-token energy (0.83 Wh/1000 tokens) is the second worst in the evaluation, surpassed only by Falcon-Mamba (0.87). Conversely, Phi-4-mini-instruct emerges as the per-token efficiency winner at 0.42 Wh/1000 tokens, with Phi-3-mini close behind (0.53). A Friedman test on $E_{1000\text{tok}}$ across the five models with the thirteen languages as blocks yields

Table 3: Generation verbosity and per-token energy across 13 languages (mean ± std). Token counts use a UTF-8-bytes/4 cross-lingual proxy. Per-query Wh/1000 queries (left, repeated from Table 2) is provided for direct comparison. Models are sorted by per-token efficiency; note the reordering relative to Table 2.

Model	Avg tokens/response	Wh/1000 queries	Wh/1000 tokens	Per-query rank → per-token rank
Phi-4-mini-instruct	202 ± 33	251.3 ± 8.6	**0.42 ± 0.06**	2 → **1**
Phi-3-mini-128k-instruct	176 ± 28	272.1 ± 6.3	0.53 ± 0.09	3 → 2
Qwen2-7B-Instruct	211 ± 23	464.6 ± 7.9	0.74 ± 0.08	5 → 3
Mistral-7B-Instruct-v0.1	64 ± 24	157.3 ± 57.4	0.83 ± 0.06	**1** → 4
Falcon-Mamba-7B	177 ± 9	461.3 ± 10.2	0.87 ± 0.04	4 → 5

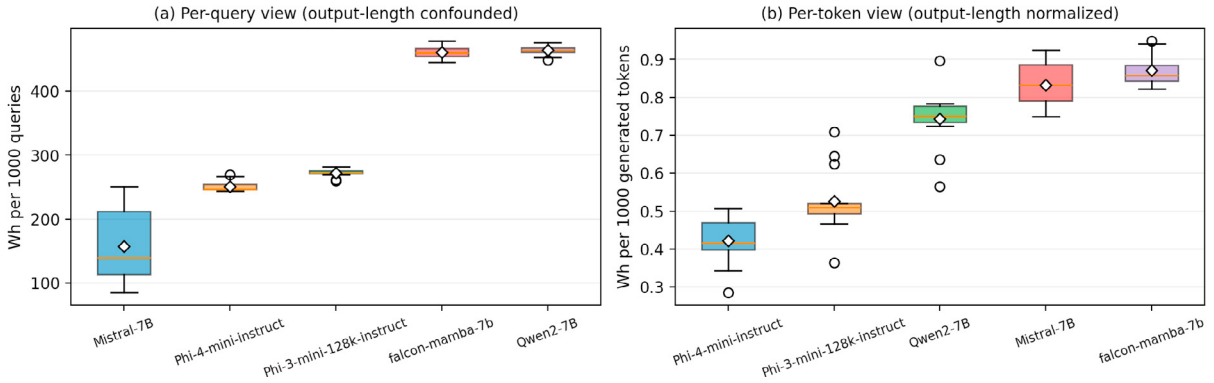


Fig. 3: Per-query (left) vs. per-token (right) energy by model, across 13 languages. The left panel reproduces the per-query view; the right panel shows the same data normalized by total generated tokens. The low-energy end of the ranking reverses: Mistral moves from rank 1 (best) to rank 4 once output length is controlled, while Phi-4-mini-instruct becomes the per-token efficiency leader.

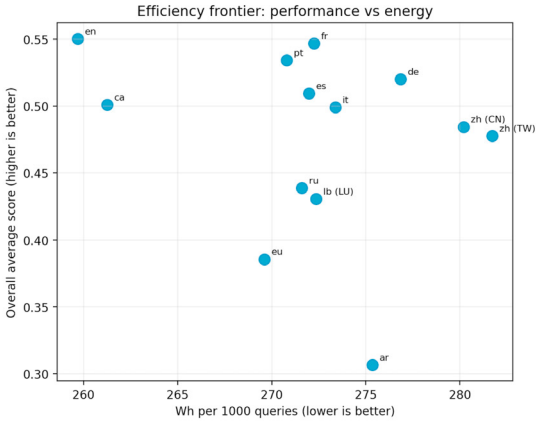
χ^2=48.31, p=8.14×10^{-10}, and all pairwise Wilcoxon–Holm contrasts are significant except Mistral vs. Falcon-Mamba (p_{Holm}=0.15).

This reordering changes the interpretation of the AI Energy Score: Mistral's high score is largely *operational* (it answers tersely, sometimes incompletely on harder problems) rather than *architectural* (per-token efficiency). Phi-4-mini-instruct, by contrast, is genuinely architecturally efficient and remains a strong choice on both views. The within-model Spearman correlation between mean tokens per response and Wh per 1,000 queries is very high for Mistral (r_s=0.978, p<0.001), confirming that Mistral's apparent language-dependent efficiency variation is primarily an output-length effect, whereas for the other four models within-model token–energy correlations across languages are weak or negative, indicating that their cross-lingual energy variance is driven by tokenization fertility on the *input* prompt rather than by output verbosity.

Pareto frontier analysis

To capture the multi-objective nature of model selection, we perform Pareto frontier analysis over E_{1000} and overall task score. Figure 7 situates all 65 model–language configurations in the joint energy–performance space and highlights the non-dominated points. As shown, high-energy models do not consistently achieve higher performance, and in several cases consume substantially more energy for comparable or inferior task outcomes. The pooled Spearman correlation between E_{1000} and overall task score across all 65 configurations is weak and non-significant (r_s=0.109, p=0.386), confirming weak energy–performance coupling.

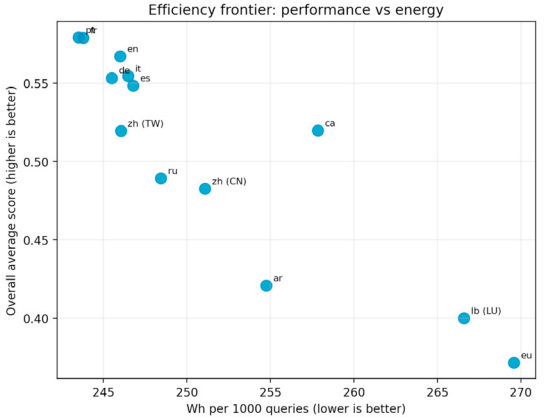
The resulting frontier shows that no individual model dominates across all objectives and languages. Mistral-7B-Instruct-v0.1 defines the low-energy extreme, offering minimal energy consumption at the cost of lower reasoning quality. Phi-4-mini-instruct occupies a favorable region of the frontier, combining high performance with moderate energy demand, while Phi-3-mini-128k-instruct provides a balanced alternative with slightly reduced accuracy. High-energy models such as Falcon-Mamba-7B and Qwen2-7B-Instruct rarely appear on the Pareto frontier, as their increased energy consumption does not yield proportional performance gains.



Energy performance by language using `Phi-3-mini-128k-instruct`

language	gpu_total_wh	wh_per_1000_queries	gpu_avg_watts	gpu_max_watts	wall_time_sec	wh_per_prompt
en	20.52	259.69	268.60	286.93	274.89	0.2597
ar	21.75	275.34	267.27	287.29	292.92	0.2753
ca	20.64	261.26	265.98	284.84	279.34	0.2613
de	21.87	276.85	267.65	286.58	294.12	0.2769
es	21.49	271.97	266.83	285.15	289.84	0.2720
eu	21.30	269.60	267.38	284.98	286.74	0.2696
fr	21.51	272.24	268.21	287.19	288.62	0.2722
it	21.60	273.38	267.00	284.82	291.14	0.2734
lb	21.52	272.37	266.96	286.57	290.07	0.2724
pt	21.39	270.80	267.45	285.82	287.97	0.2708
ru	21.46	271.59	266.00	284.54	290.31	0.2716
zh-cn	22.14	280.20	270.13	288.20	294.93	0.2802
zh-tw	22.26	281.71	270.59	288.28	296.05	0.2817

Fig. 4: Energy–performance frontier for Phi-3-mini-128k-instruct, illustrating a balanced trade-off across multilingual configurations.



Energy performance by language using `Phi-4-mini-instruct`

language	gpu_total_wh	wh_per_1000_queries	gpu_avg_watts	gpu_max_watts	wall_time_sec	wh_per_prompt
en	19.43	246.00	258.34	278.82	270.78	0.2460
ar	20.13	254.75	255.45	279.03	283.56	0.2547
ca	20.37	257.84	257.93	283.00	284.25	0.2578
de	19.40	245.51	258.24	278.80	270.30	0.2455
es	19.50	246.77	258.42	279.55	271.51	0.2468
eu	21.30	269.57	258.80	279.09	296.22	0.2696
fr	19.26	243.78	257.35	279.53	269.40	0.2438
it	19.47	246.48	256.18	278.63	273.61	0.2465
lb	21.06	266.58	259.12	279.67	292.58	0.2666
pt	19.24	243.52	258.76	280.22	267.56	0.2435
ru	19.63	248.45	258.20	279.47	273.56	0.2484
zh-cn	19.84	251.08	259.80	279.54	274.78	0.2511
zh-tw	19.44	246.04	258.72	278.69	270.43	0.2460

Fig. 5: Energy–performance frontier for Phi-4-mini-instruct. High reasoning performance at moderate energy cost.

4. Discussion

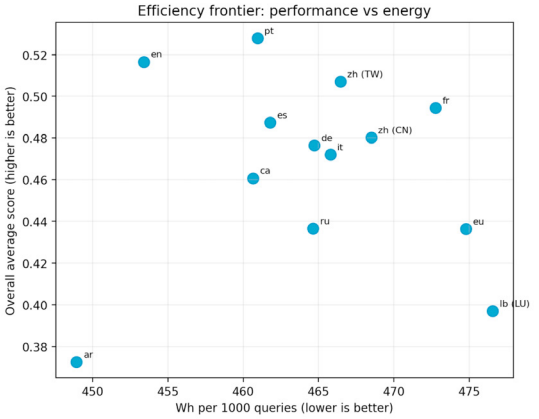
The observed decoupling between energy consumption and reasoning performance challenges the assumption that higher computational cost yields proportionally better task outcomes, motivating efficiency-aware evaluation for sustainable deployment.

A key finding is the emergence of distinct energy–performance regimes associated with architectural strategies, and the further reordering revealed by the per-token analysis (Section 3): the apparent low-energy advantage of Mistral-7B is largely an artifact of short output length, whereas Phi-4-mini-instruct is the genuine architectural efficiency leader. This decomposition matters for deployment planning, since systems with adaptive routing or chain-of-thought elaboration will pay the per-token cost, not the per-query cost calibrated on short outputs. Practitioners should therefore consider both views: E_{1000} reflects *operational* cost under the model's current generation behavior, while $E_{1000\text{tok}}$ reflects *architectural* efficiency that is robust to changes in prompting style or output budget.

The multilingual dimension amplifies these effects: language choice shifts both Pareto-optimality and relative model efficiency, so English-only benchmarks do not generalize to multilingual deployment. Romance languages (ca, es, pt) on average yield lower energy than English, suggesting language itself is a tunable deployment parameter.

Model selection is therefore a multi-objective decision: Phi-4-mini-instruct occupies a favorable region for accuracy-sensitive workloads, while Mistral-7B-Instruct-v0.1 suits energy-constrained settings where approximate answers are acceptable.

The energy differences observed at our experimental scale translate into substantial operational and environmental costs at deployment scale. At a serving load of 1 million queries/day and a typical data-centre electricity price of \$0.10/kWh, the gap between the low-energy regime (Mistral, ~157 Wh/1000q) and the high-energy regime (Qwen2, Falcon-Mamba, ~462 Wh/1000q) corresponds to roughly \$11,200/year purely in electricity cost; at 10 million queries/day the difference reaches \$112,000/year and approximately 261 tCO_2/year on an average EU grid (car-



Energy performance by language using `Qwen2-7B-Instruct`.

language	gpu_total_wh	wh_per_1000_queries	gpu_avg_watts	gpu_max_watts	wall_time_sec	wh_per_prompt
en	35.82	453.40	325.29	346.91	396.31	0.4534
ar	35.47	448.94	325.21	347.85	392.55	0.4489
ca	36.39	460.65	327.09	349.72	400.46	0.4607
de	36.71	464.72	325.45	347.10	406.10	0.4647
es	36.48	461.77	327.52	348.39	400.97	0.4618
eu	37.51	474.78	327.29	346.64	412.54	0.4748
fr	37.35	472.75	328.11	348.48	409.68	0.4728
it	36.80	465.78	325.71	345.34	406.63	0.4658
lb	37.65	476.54	325.59	348.43	416.16	0.4765
pt	36.41	460.94	325.68	347.42	402.51	0.4609
ru	36.71	464.64	324.67	348.97	406.91	0.4646
zh-cn	37.01	468.49	325.38	347.72	409.41	0.4685
zh-tw	36.85	466.44	326.08	348.63	406.76	0.4664

Fig. 6: Energy–performance frontier for Qwen2-7B-Instruct, highlighting high energy without proportional performance gains.

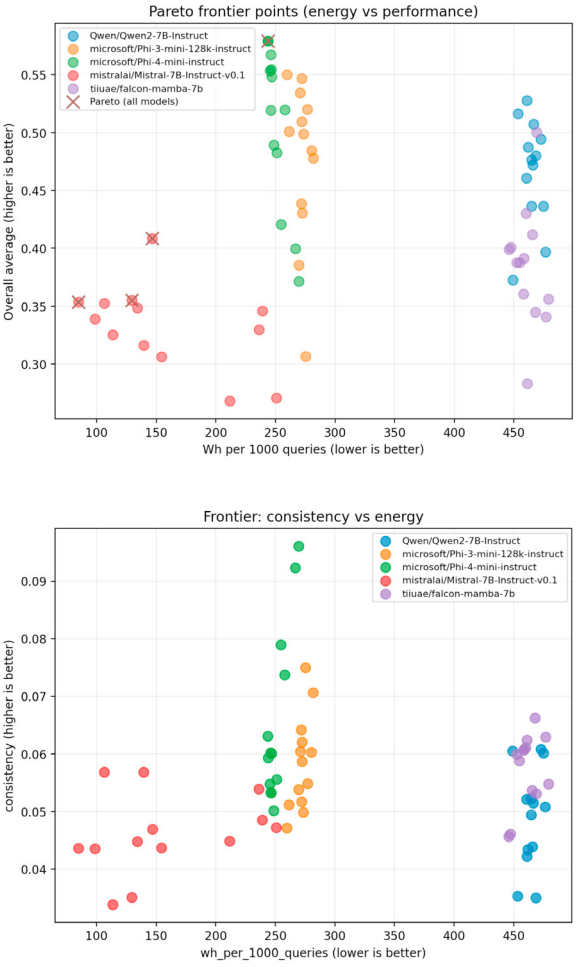


Fig. 7: Global Pareto frontier across all models and languages. Points on the frontier jointly minimise energy and maximise reasoning performance.

bon intensity 0.233 $kgCO_2/kWh$). When language-aware routing is also applied (e.g. routing Portuguese inputs to a Pareto-optimal configuration), additional savings of ~$13,700/year at 1 million queries/day are achievable. These figures motivate the use of AI Energy Scores as a complementary criterion in sustainability-aware procurement and model-selection workflows.

Several caveats apply. First, all experiments were conducted on a single hardware platform (NVIDIA H200), one inference framework, and a fixed decoding configuration; different accelerators, batch sizes, or sampling strategies may alter absolute energy values and could in principle reorder the rankings, although the magnitudes of the observed effects suggest the qualitative conclusions would persist. Second, the evaluation set comprises 79 multi-step reasoning problems; while modest relative to large-scale accuracy benchmarks, the paired-blocks statistical framework on 13 languages provides 80% power for paired effect sizes $d \geq 0.78$, and all of the pairwise effects reported here are well above this threshold (most $|d| > 2$), so the conclusions are statistically robust on this benchmark and would be expected to generalize provided benchmark composition does not differ drastically. Third, only five open-weight models in the 3.8–7.3 B parameter range are evaluated; the compact-model advantage observed here should not be extrapolated to substantially larger architectures without further empirical validation. Fourth, the per-token analysis uses a UTF-8-bytes/4 cross-lingual proxy for token counts because the per-model tokenizers were not available at analysis time;

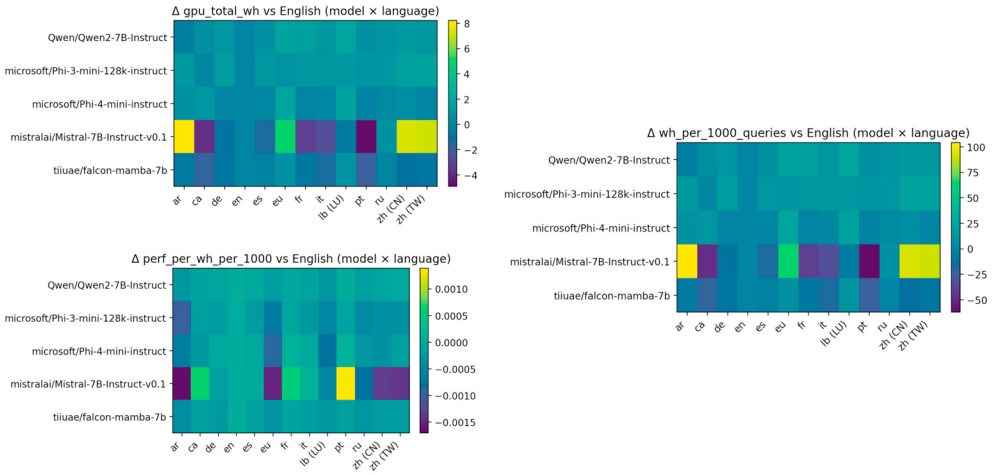


Fig. 8: Language-dependent variation relative to the English baseline across all evaluated models. Left column: deviation in total GPU energy consumption (top) and performance efficiency (overall score per Wh per 1,000 queries, bottom). Right column: deviation in energy intensity (Wh per 1,000 queries). Warmer colours indicate higher values relative to English.

absolute Wh/1000-token values should be interpreted as proportional rather than exact, but the per-model and per-language ratios on which our conclusions rely are robust to any multiplicative scaling of the token count. Finally, this study focuses exclusively on inference-time behavior and does not consider training-related energy costs.

5. Conclusions

We presented a systematic multilingual evaluation of five instruction-tuned Large Language Models across thirteen languages, jointly analyzing GPU-level energy and reasoning performance. Energy and task accuracy are only weakly coupled, and higher energy does not reliably translate into superior reasoning quality.

Beyond the per-query analysis common in prior energy-aware LLM studies, we introduced an output-length-normalized energy view that decomposes operational verbosity from architectural per-token efficiency. This view reorders the efficiency ranking: the apparently most-efficient model under the per-query metric (Mistral-7B, on account of short outputs) becomes second-worst on the per-token metric, while compact Phi-family Transformers with Grouped-Query Attention are the genuine architectural-efficiency leaders. Furthermore, language choice was shown to influence both energy demand and relative model efficiency, leading to language-dependent Pareto-optimal configurations and unstable global rankings.

These results have direct implications for the sustainable deployment of LLM-based systems. They suggest that model selection should be guided by multi-objective criteria that explicitly account for energy consumption, performance requirements, and linguistic coverage, rather than relying solely on accuracy-centric benchmarks. AI Energy Scores, especially when reported in both per-query and per-token forms, provide a useful complementary perspective for comparing models under real-world operational constraints.

Future work will extend this evaluation framework to additional hardware platforms, decoding strategies, and task domains, replace the bytes/4 token proxy with native per-model tokenizers for more precise per-token analysis, and explore adaptive deployment strategies that dynamically balance energy efficiency and performance based on workload and language.

Acknowledgements

This research was supported by the LUXEMBOURG Institute of Science and Technology through the projects ‘ADIALab-MAST’ and ‘LLMs4EU’ (Grant Agreement No 101198470) and the BARCELONA Supercomputing Center through the project ‘TIFON’ (File number MIG-20232039).

Code Availability

The complete implementation is publicly available at https://github.com/drdecurto/AI-energy-scores-multilingual, including multilingual data handling, GPU energy measurement aggregation, and the per-token reanalysis notebook.

References

[1] OpenAI, GPT-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[2] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al., Llama 2: Open foundation and fine-tuned chat models, arXiv preprint arXiv:2307.09288 (2023).

[3] Anthropic, Claude 3 model card (2024).
URL https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf

[4] J. de Curtò, I. de Zarzà, C. T. Calafate, Llm multi-agent decision optimization, in: G. Jezic, Y.-H. Chen-Burger, M. Kušek, R. Šperka, R. J. Howlett, L. C. Jain (Eds.), Agents and Multi-agent Systems: Technologies and Applications 2024, Springer Nature Singapore, Singapore, 2025, pp. 3–15.

[5] R. Schwartz, J. Dodge, N. A. Smith, O. Etzioni, Green ai, Communications of the ACM 63 (12) (2020) 54–63.

[6] E. Strubell, A. Ganesh, A. McCallum, Energy and policy considerations for deep learning in nlp, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (2019) 3645–3650.

[7] D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, J. Steinhardt, Measuring mathematical problem solving with the math dataset, arXiv preprint arXiv:2103.03874 (2021).

[8] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, et al., Beyond the imitation game: Quantifying and extrapolating the capabilities of language models, Transactions on Machine Learning Research (2023).

[9] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, J. Pineau, Towards the systematic reporting of the energy and carbon footprints of machine learning, Journal of Machine Learning Research 21 (248) (2020) 1–43.

[10] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. H. Chi, Q. V. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, in: Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Curran Associates Inc., Red Hook, NY, USA, 2022.

[11] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, D. Zhou, Self-consistency improves chain of thought reasoning in language models, arXiv preprint arXiv:2203.11171 (2023).

[12] A. Lacoste, A. Luccioni, V. Schmidt, T. Dandres, Quantifying the carbon emissions of machine learning, arXiv preprint arXiv:1910.09700 (2019).

[13] P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayanan, Y. Wu, A. Kumar, et al., Holistic evaluation of language models, arXiv preprint arXiv:2211.09110 (2022).

[14] L. Zhou, W. Schellaert, F. Martínez-Plumed, Y. Moros-Daval, C. Ferri, J. Hernández-Orallo, Larger and more instructable language models become less reliable, Nature 634 (8032) (2024) 61–68.

[15] Y. Elazar, N. Kassner, S. Ravfogel, A. Ravichander, E. Hovy, H. Schütze, Y. Goldberg, Measuring and improving consistency in pretrained language models, Transactions of the Association for Computational Linguistics 9 (2021) 1012–1031.

[16] J. Wang, X. Hu, W. Hou, H. Chen, R. Zheng, Y. Wang, L. Yang, H. Huang, W. Ye, X. Geng, et al., On the robustness of ChatGPT: An adversarial and out-of-distribution perspective, arXiv preprint arXiv:2302.12095 (2023).

[17] I. de Zarzà, M. Liz, J. de Curtò, C. T. Calafate, Energy-aware multilingual evaluation of large language models, Electronics 15 (7) (2026). doi:10.3390/electronics15071395.

[18] W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, I. Stoica, Efficient memory management for large language model serving with pagedattention, in: Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles, 2023.

[19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., PyTorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems, Vol. 32, 2019.

[20] J. de Curtò, I. de Zarzà, Metamorphic testing for semantic invariance in large language models, IEEE Access 13 (2025) 214772–214791. doi:10.1109/ACCESS.2025.3646270.

[21] E. M. Bender, A. Koller, Climbing towards nlu: On meaning, form, and understanding in the age of data, in: Proceedings of the 58th annual meeting of the association for computational linguistics, 2020, pp. 5185–5198.

[22] Hugging Face, S. Luccioni, Y. Jernite, R. Pierrard, I. Moutawwakil, M. Mitchell, B. Gamazaychikov, S. Chamberlin, S. Hooker, C.-J. Wu, E. Strubell, AI Energy Score: Standardized energy efficiency ratings for AI models, accessed: February 2026 (2025).
URL https://huggingface.github.io/AIEnergyScore/

[23] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, et al., Training verifiers to solve math word problems, arXiv preprint arXiv:2110.14168 (2021).

[24] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: Nsga-ii, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182–197.

[25] A. Petrov, E. La Malfa, P. Torr, A. Bibi, Language model tokenizers introduce unfairness between languages, Advances in neural information processing systems 36 (2023) 36963–36990.